Fig. 1

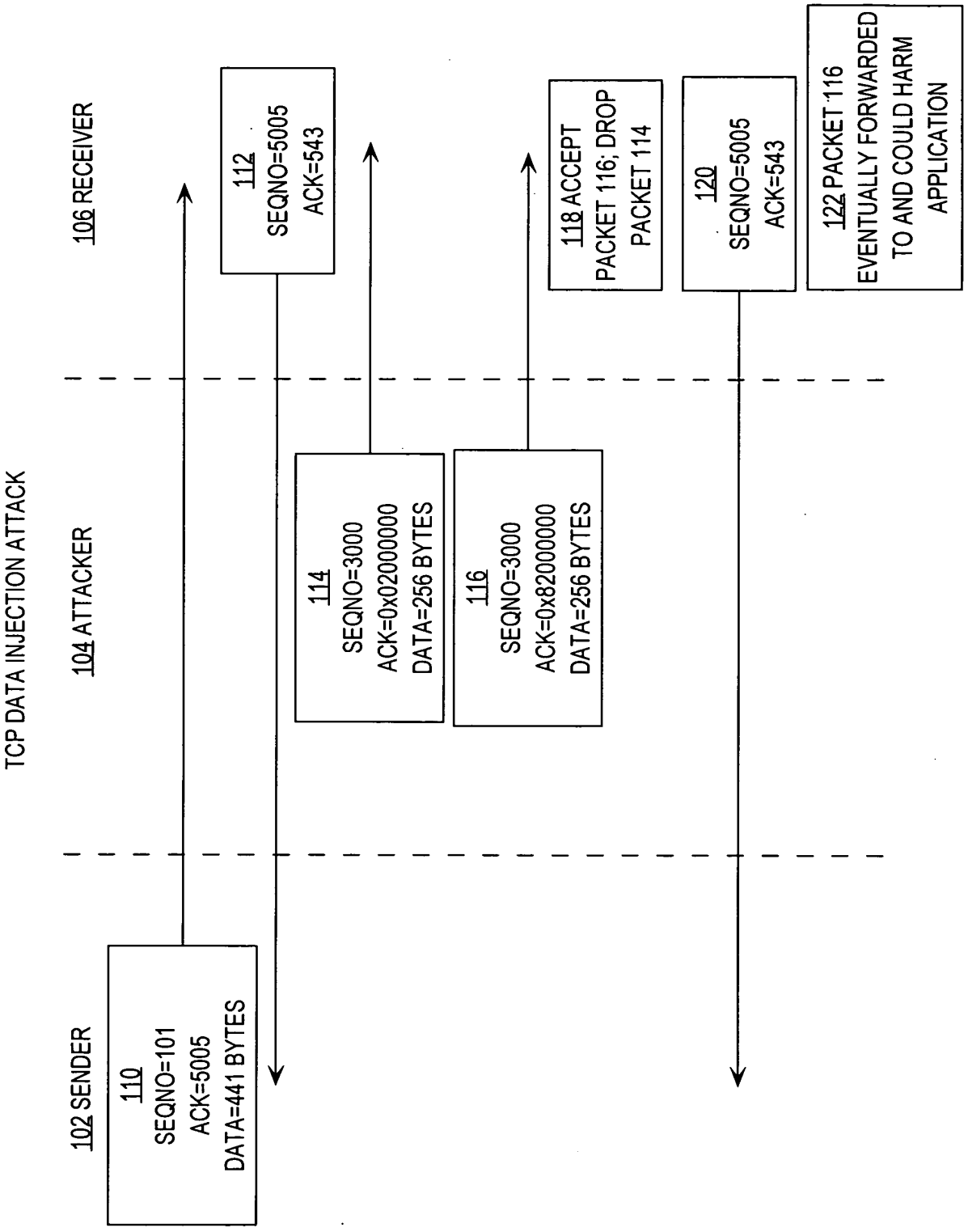

TCP DATA INJECTION ATTACK
102 SENDER
104 ATTACKER
106 RECEIVER
110 SEQNO=101 ACK=5005 DATA=441 BYTES
112 SEQNO=5005 ACK=543
114 SEQNO=3000 ACK=0x02000000 DATA=256 BYTES
116 SEQNO=3000 ACK=0x82000000 DATA=256 BYTES
118 ACCEPT PACKET 116; DROP PACKET 114
120 SEQNO=5005 ACK=543
122 PACKET 116 EVENTUALLY FORWARDED TO AND COULD HARM APPLICATION

Fig. 2

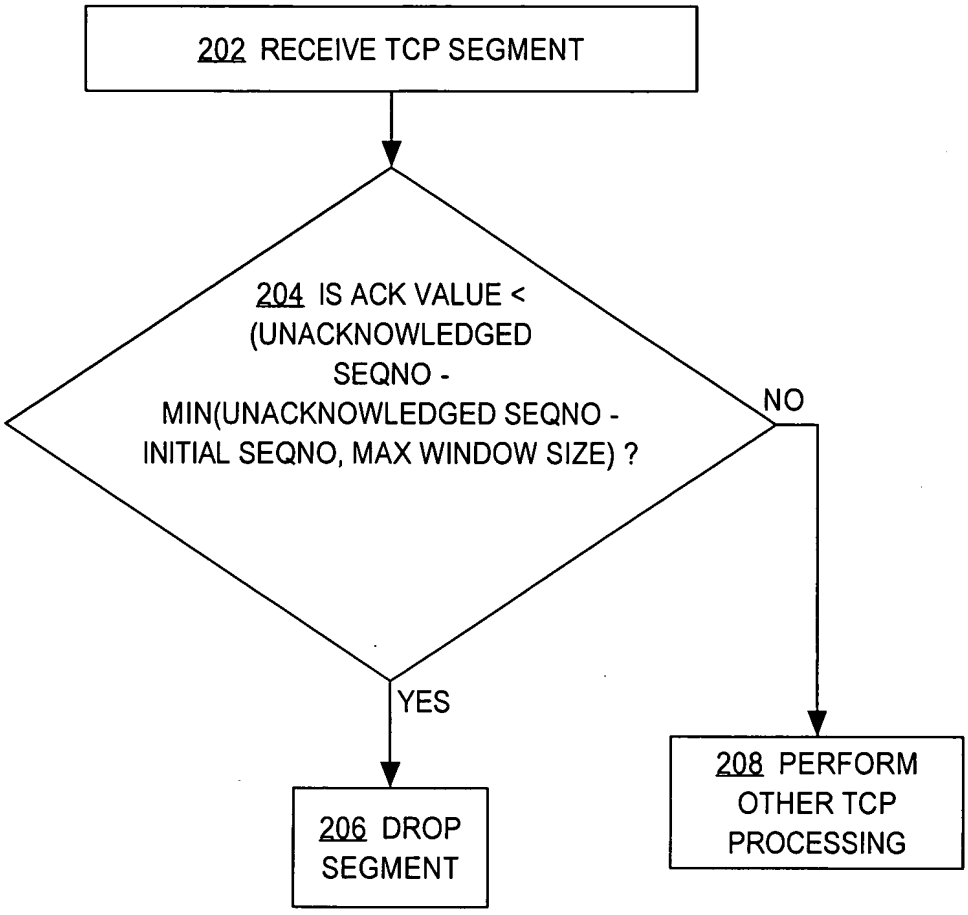

202 RECEIVE TCP SEGMENT
204 IS ACK VALUE < (UNACKNOWLEDGED SEQNO - MIN(UNACKNOWLEDGED SEQNO - INITIAL SEQNO, MAX WINDOW SIZE) ?
NO
YES
206 DROP SEGMENT
208 PERFORM OTHER TCP PROCESSING

Fig. 3

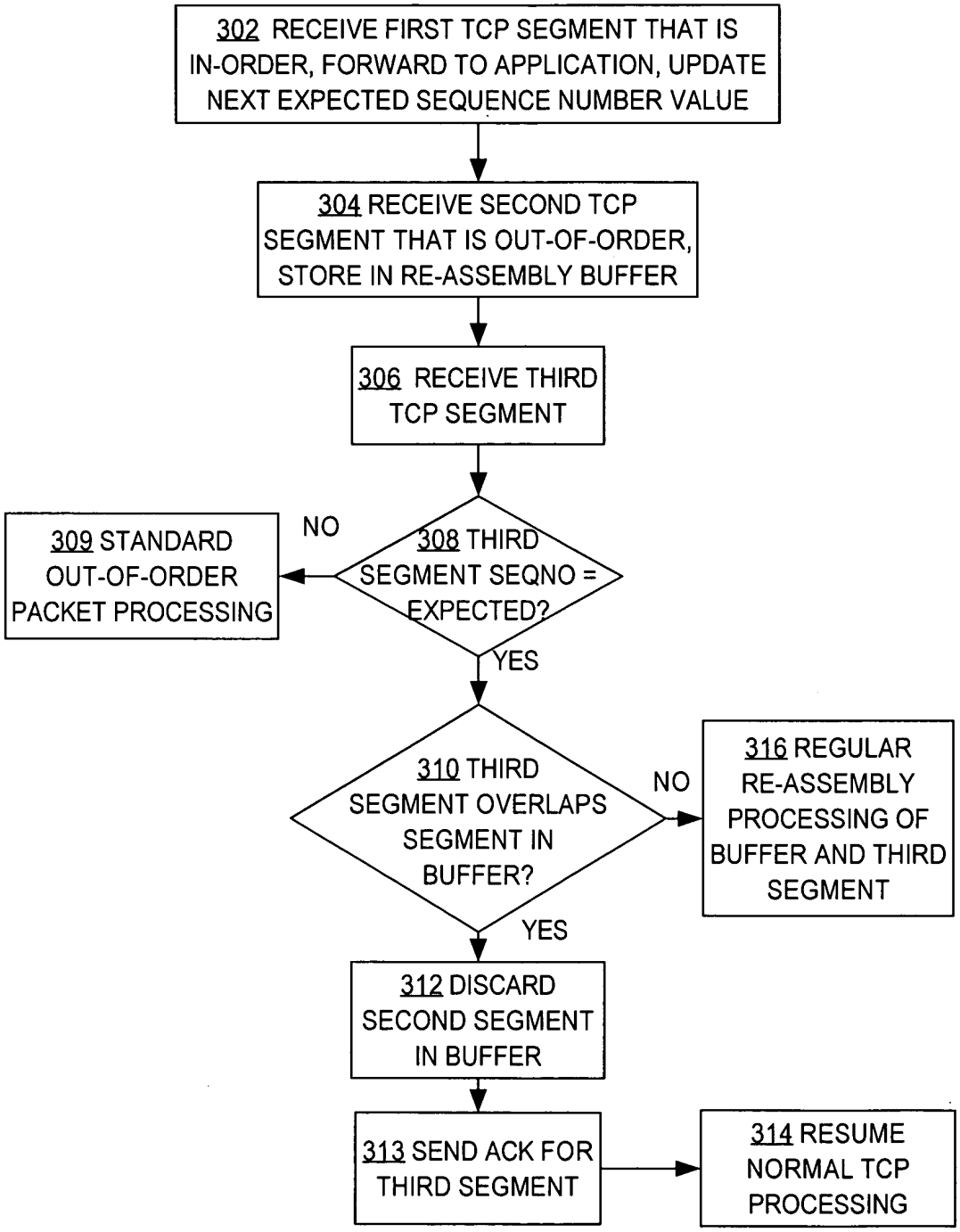
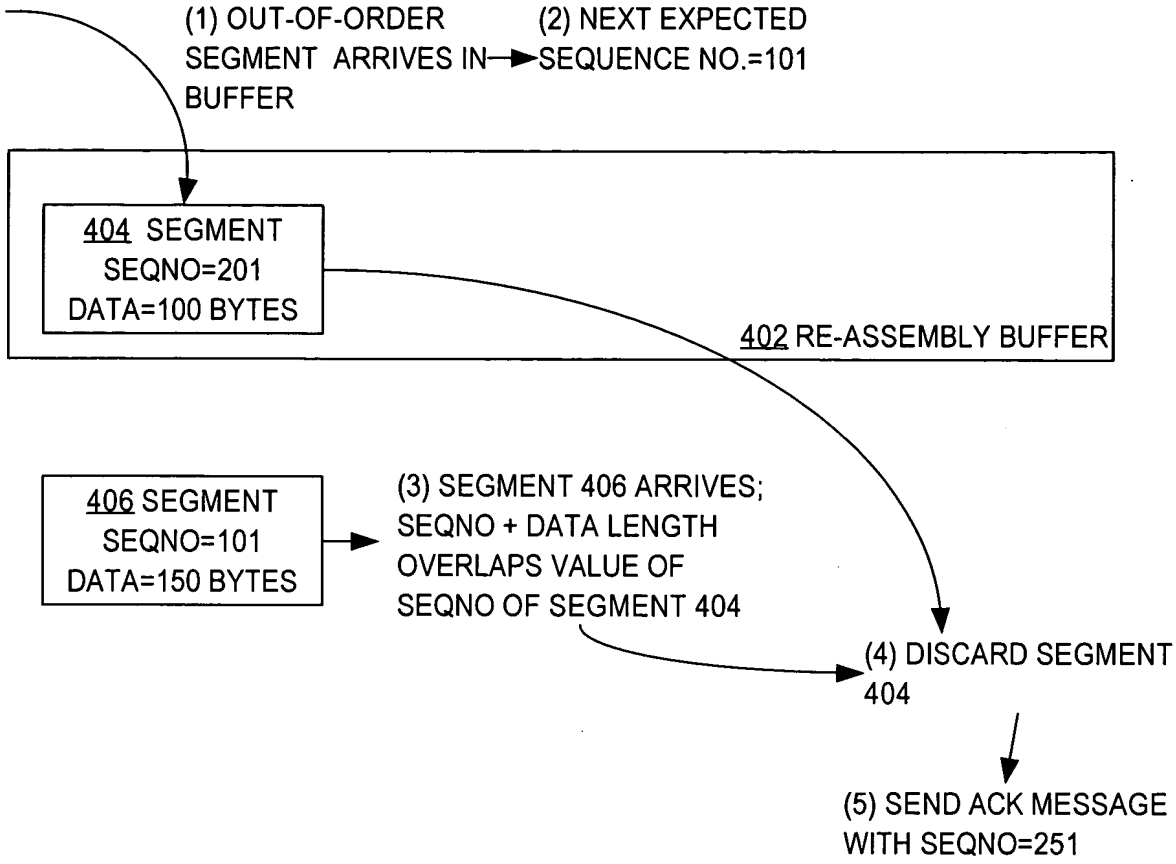

302 RECEIVE FIRST TCP SEGMENT THAT IS IN-ORDER, FORWARD TO APPLICATION, UPDATE NEXT EXPECTED SEQUENCE NUMBER VALUE
304 RECEIVE SECOND TCP SEGMENT THAT IS OUT-OF-ORDER, STORE IN RE-ASSEMBLY BUFFER
306 RECEIVE THIRD TCP SEGMENT
308 THIRD SEGMENT SEQNO = EXPECTED?
NO
309 STANDARD OUT-OF-ORDER PACKET PROCESSING
YES
310 THIRD SEGMENT OVERLAPS SEGMENT IN BUFFER?
NO
316 REGULAR RE-ASSEMBLY PROCESSING OF BUFFER AND THIRD SEGMENT
YES
312 DISCARD SECOND SEGMENT IN BUFFER
313 SEND ACK FOR THIRD SEGMENT
314 RESUME NORMAL TCP PROCESSING

Fig. 4


(1) OUT-OF-ORDER SEGMENT ARRIVES IN BUFFER
(2) NEXT EXPECTED SEQUENCE NO.=101
404 SEGMENT SEQNO=201 DATA=100 BYTES
402 RE-ASSEMBLY BUFFER
406 SEGMENT SEQNO=101 DATA=150 BYTES
(3) SEGMENT 406 ARRIVES; SEQNO + DATA LENGTH OVERLAPS VALUE OF SEQNO OF SEGMENT 404
(4) DISCARD SEGMENT 404
(5) SEND ACK MESSAGE WITH SEQNO=251

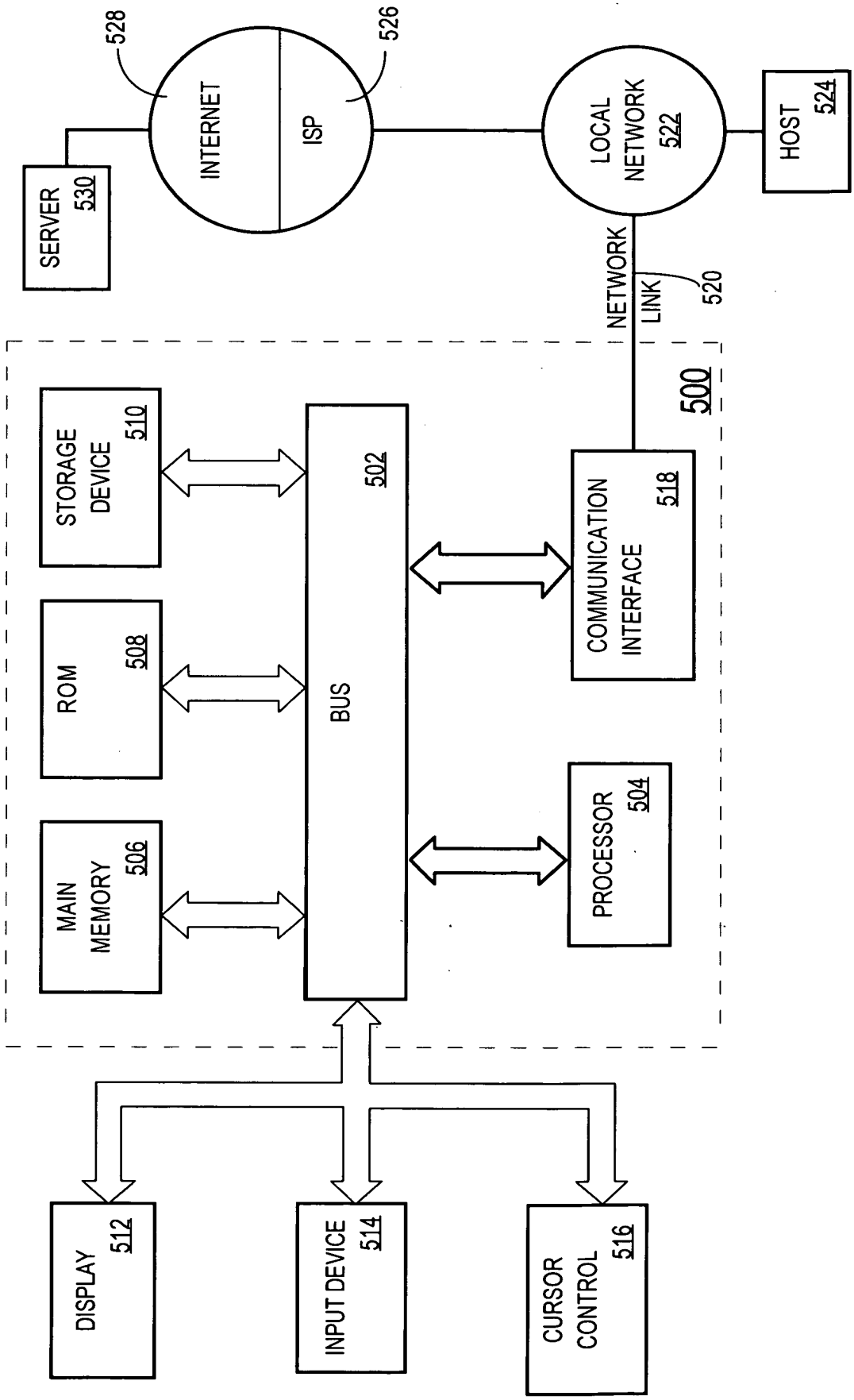

528
INTERNET
ISP
526
SERVER
530
LOCAL
NETWORK
522
HOST
524
NETWORK
LINK
520
500
STORAGE
DEVICE
510
502
COMMUNICATION
INTERFACE
518
ROM
508
BUS
MAIN
MEMORY
506
PROCESSOR
504
DISPLAY
512
INPUT DEVICE
514
CURSOR
CONTROL
516

Fig. 5